



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/042,400	01/09/2002	Laura J. Poplawski	END920010032US1(14761)	9037

7590 07/07/2006

Steven Fischman, Esq.  
Scully, Scott, Murphy & Presser  
400 Garden City Plaza  
Garden City, NY 11530

EXAMINER
----------

LERNER, MARTIN

ART UNIT	PAPER NUMBER
----------	--------------

2626

DATE MAILED: 07/07/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	10/042,400	POPLAWSKI, LAURA J.	
	<b>Examiner</b>	<b>Art Unit</b>	
	Martin Lerner	2626	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 22 May 2006.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1, 3 to 6, 8 to 11, 13 to 15, and 17 to 19 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1, 3 to 6, 8 to 11, 13 to 15, and 17 to 19 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |   |   |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)             | 4) <input type="checkbox"/> Interview Summary (PTO-413)                     |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)    | Paper No(s)/Mail Date. _____  |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____   | 6) <input type="checkbox"/> Other: _____                                    |

## DETAILED ACTION

### ***Claim Rejections - 35 USC § 112***

The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 18 and 19 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claims contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor, at the time the application was filed, had possession of the claimed invention.

Claim 18 contains the limitations of "the step of the user creating said map file includes the step of the user specifying in said map file which elements of the legacy file are to have multiple occurrences in said markup language file, and which elements of the legacy file are to be nested in the markup language file", which limitations involve new matter. Applicant's Specification, Page 8, Lines 6 to 9, as originally filed, only states, "The parser can deal with optional elements, multiple occurrence elements, nested elements, PCDATA anywhere within an element (including after a child element), attributes, doctypes, processing instructions." However, Applicant's Specification, as originally filed, does not disclose a user creating the map file and specifying that the map file have elements of the legacy file that are to have multiple

occurrences in the markup language, and elements of the legacy file that are to be nested in the markup language. Applicant's Specification only hints that a parser can deal with these elements, but does not disclose anything about them being created by a user and specified in a map file for a markup language. Any multiple occurrences and nested elements could simply be present in a legacy file, without any references being specified and created by a user in a map file for the markup language.

### ***Claim Rejections - 35 USC § 102***

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1, 5, 6, 10, 11, and 15 are rejected under 35 U.S.C. 102(e) as being anticipated by *Ballantyne et al.*

Regarding independent claims 1, 6, and 11, *Ballantyne et al.* discloses a method, system, and program instructions for converting XML data from a legacy computer system, comprising;

“providing a delimited flat legacy file having a plurality of columns with text, each of said columns having a column heading” – a “flat file” is a simple database model, where information is stored in a plain text file, with one database record per line, each

Art Unit: 2626

record being divided into fields using delimiters at fixed column positions (*Wikipedia*);

Figure 4 illustrates a flat file from COBOL legacy code, with one record per line, columns and headings for date, time, number, city, duration, cost, etc., where each column is delimited by fields; text data of flat file records includes cities "San Antoni", "Kill Devil", etc. (column 8, lines 46 to 58: Figure 4);

"providing a map file conforming to said document type definition file and having tags and attributes including references matching said headings, wherein each of the column headings is matched by one of the references included in said attributes" – modeling/mapping graphical user interface 30 illustrates the mapping relationship between the XML schema, the report data model, and the underlying legacy computer program application depicted as COBOL (column 10, lines 4 to 22: Figures 4 to 6); the mapping relationship is a program defining a mapping engine 24 for creating modified legacy program applications (column 10, lines 54 to 65); a mapping is defined by attributes and tags for XML to match reference headings in COBOL (column 12, lines 11 to 45); a "document type definition file" is bit of markup that appears near the start of an XML document, and establishes that the document is an instance of a referenced type (*Wikipedia*); thus, markup establishing a schema at the beginning of the XML documents of Figures 5A and 7 are "document type definitions", *i.e.* <Schema xmlns = "urn: schemas-microsoft-com:xml-data" . . .>;

"forming a tree structure from said map file for mapping said text from said flat file into a defined format in said markup language file, and wherein each tag represents one or more nodes of said tree" – a data structure for an XML schema is a tree structure of

Art Unit: 2626

elements (column 11, lines 29 to 47: Figures 7 and 7A); elements correspond to tags for XML; elements of a tree structure include text elements for “city”;

“reading the map file and using the map file to map text from the legacy file into the defined format of said markup language file, including the step of traversing said legacy file, column by column, and for each of the columns, mapping all of the text in the column to said markup language file, including the steps of: traversing said nodes of said tree structure, node-by-node, and for each said node entering said attributes into said markup language file; traversing all of said nodes of said tree to ensure that references are found matching all of the column headings of the legacy file, and thereby to ensure that all of the text from the legacy file is retrieved therefrom and entered into the markup language file” — a tree structure is utilized for rewriting a legacy program code from COBOL into XML (“said markup language file”) by traversing the elements of a tree structure for each element (column 11, lines 29 to 47: Figures 7 and 7A); thus, text elements for “address”, “city”, etc., are obtained from a source program for a target program by traversing every node of a tree structure of Figure 7A; the mapping engine generates the modification specification and context table by mapping a model of write operations of the legacy computer program to an XML schema (column 3, lines 12 to 27); implicitly, nodes of a tree mapping from a legacy file to an XML schema are written “column by column” and “node-by-node” “for each node of said tree”, and “ensure that references are found matching all of the column headings” “to ensure that all of the text . . . is retrieved” as part of the process of rewriting a COBOL legacy application into an XML format with a tree structure of Figure 7A.

Regarding claims 5, 10, and 15, *Ballantyne et al.* discloses a legacy file in COBOL, which is a flat file delimited by tabs defining columns for date, time, number, city and state, duration, cost, etc. (Figure 4).

### ***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 3, 4, 8, 9, 13, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Ballantyne et al.* in view of *Baisley et al.*

*Ballantyne et al.* omits the steps of providing a map file for default text for certain elements and attributes in the markup language, and entering the default text into the markup language for attributes having references that do not match headings of the flat file. Ordinarily, it would be presumed that all corresponding elements of matching between a source flat file and an object file are provided, but it is well known that there are exceptional instances where they may not, whereupon a default procedure must be specified. (Analogously, when a file name is not specified for saving the file in Windows®, an opening text segment of a file is designated as a default file name.)

*Baisley et al.* teaches a procedure for converting from one modeling language to another, wherein object models existing in a Uniform Modeling Language (UML) are converted to models existing in a Meta Object Facility Language (MOF). Specifically, it

Art Unit: 2626

is stated that it is not always possible to generate a name for each unnamed element, and generated names often do not serve the purpose of describing the named element. Thus, when no name is provided, or when a name is omitted from both ends, the end's type may be a suitable name, a numeral may be appended to an offending name that violates a rule constraint of UML, or it may be given the name "Contains". (Column 4, Line 49 to Column 5, Line 67) The objective is to provide a set of rules for making a transformation between models in object-oriented programming languages with a predictable mapping. (Column 1, Lines 39 to 67) It would have been obvious to one having ordinary skill in the art to apply the default naming conventions taught by *Baisley et al.* in the method and system for modifying legacy programs into XML of *Ballantyne et al.* for the purpose of providing transformation rules between programming languages with a predictable mapping.

Claims 17 to 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Ballantyne et al.* in view of *Morganstern*.

*Ballantyne et al.* discloses a legacy file having column headings for COBOL legacy code (Figure 4), but does not expressly disclose that a user creates the map file, that a user provides the map file with references, or that the map file specifies elements of the legacy file that are to have multiple occurrences or are to be nested in the markup language. However, *Morganstern* teaches an integration platform for heterogeneous databases, where an integration administrator 14 provides an information bridge 1. (Column 5, Lines 61 to 67) The information bridge is a method for integrating



heterogeneous data from source data to target data (column 2, lines 60 to 64), where XML is integrated from legacy HTML (column 38, lines 15 to 21; column 45, lines 28 to 32). Thus, an integration administration is “a user” who performs the integration. A schematic structure graph, or “tree”, is created, as a logical structure diagram (LSD) for relations, attributes, or objects having a set of nodes and edges. (Column 11, Lines 25 to 39; Column 12, Lines 3 to 42) Thus, a logical structure diagram (LSD), created by an integration administrator, is a “map file” for integrating diverse databases by a graphical tree structure of data entities. Moreover, *Morganstern* states that a logical structure diagram (LSD) may contain hierarchically nested data (column 11, lines 40 to 42), multiple data instances or occurrences signified by a “+” (column 24, lines 22 to 26; column 24, lines 63 to 65), and a basic group of the logical model is a metaframe with sub-attributes that may be nested arbitrarily deep (column 39, line 50 to column 40, line 29). The objective is to provide interoperability for heterogeneous databases. (Column 2, Line 59 to Column 3, Line 34) It would have been obvious to one having ordinary skill in the art for a user to create a map file with references to elements that have multiple occurrences or are to be nested as taught by *Morganstern* in a method and system for reporting XML data from a legacy computer system of *Ballantyne et al.* for a purpose of providing interoperability for heterogeneous databases.

### ***Response to Arguments***

Applicant's arguments filed 22 May 2006 have been fully considered but they are not persuasive.

Firstly, Applicant argues that the claimed method, system, and program converts a complete legacy file to a specified markup language, while *Ballantyne et al.* is directed to outputting legacy data in an XML format. Applicant says that they employ a procedure to ensure that all of the text of the legacy file is retrieved and converted, but no proactive assurance that a complete converted legacy file is provided by *Ballantyne et al.* Applicant's argument is not persuasive.

*Ballantyne et al.* describes a procedure that would necessarily ensure that all of the text of the legacy file is completely retrieved and converted. Applicant has pointed to nothing that would tend to show that less than all of the text is converted in *Ballantyne et al.* Applicant is merely stating a bare allegation. The objective of mapping by a tree structure is to report all of the data from a legacy program to XML for *Ballantyne et al.*

Secondly, Applicant cites *Ballantyne et al.* as disclosing providing "XML output by modifying the underlying legacy computer system program applications to report data in XML format instead of transforming the output from the legacy computer system after the data is reported in the format of the legacy computer system." Thus, Applicant concludes that *Ballantyne et al.* teaches an opposite approach, and actually teaches away from the claimed method, system, and program. This position is not persuasive.

Applicant admits that *Ballantyne et al.* modifies the legacy computer application so that the application outputs the data in the desired way. It is unclear how Applicant believes the claimed method, system, and program differs from that of *Ballantyne et al.*, or why *Ballantyne et al.* is opposite and teaches away. *Ballantyne et al.* says it modifies the program application to report legacy data in XML format instead of transforming the output after it is reported. By implication, Applicant's cited passage states that the program is modified so that the output does not have to be recompiled each time a user wants to view the same output data. However, if *Ballantyne et al.* is directed to modifying the underlying program instead of transforming the output each time it is reported, it is not apparent how that would differ from Applicant's claimed method, system, and program.

Thirdly, Applicant argues that *Ballantyne et al.* shows text that is a flat file in Figure 4, and shows this text in an XML format in Figure 5, but does not convert the text of Figure 4 to Figure 5. Instead, Applicant says that Figures 4 and 5 represent two printed outputs of the same basic data. This position is traversed.

After modifying the legacy computer application so that it outputs the data in the desired way, the data may appear in exactly the same format as it did previously or it may appear in a different format for *Ballantyne et al.* However, the relevance of Applicant's argument to the claimed method, system, and program is not clear. Applicant's claims require converting text from a legacy file to a text in a markup language by a map file and a tree structure for mapping the text from columns in the legacy file into a defined format in the markup language. Figure 4 may represent

Art Unit: 2626

displayed output of data in COBOL, and Figure 5 may represent data as stored in XML format before it is output by *Ballantyne et al.* Still, *Ballantyne et al.* provides columns of text in a legacy file, and a defined format in a markup language, as claimed. Any features upon which Applicant relies with respect to a distinction of format for Figures 4 and 5 are not recited in the rejected claims. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Therefore, the rejection of claims 18 and 19 under 35 U.S.C. §112, 1<sup>st</sup> ¶, as failing to comply with the written description requirement; of claims 1, 5, 6, 10, 11, and 15 under 35 U.S.C. §102(e) as being anticipated by *Ballantyne et al.*; of claims 3, 4, 8, 9, 13, and 14 under 35 U.S.C. §103(a) as being unpatentable over *Ballantyne et al.* in view of *Baisley et al.*; and of claims 17 to 19 under 35 U.S.C. §103(a) as being unpatentable over *Ballantyne et al.* in view of *Morganstern*, are proper.

### **Conclusion**

The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure.

Wikipedia discloses "Document Type Definition".

Makely et al. and Kuznetsov disclose related art.

Applicant's amendment necessitated the new grounds of rejection presented in the Office Action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Martin Lerner whose telephone number is (571) 272-7608. The examiner can normally be reached on 8:30 AM to 6:00 PM Monday to Thursday.

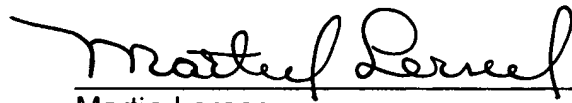
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David R. Hudspeth can be reached on (571) 272-7843. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only.

Art Unit: 2626

For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

ML  
6/29/06

A handwritten signature in black ink, reading "Martin Lerner". The signature is written in a cursive style with a horizontal line underneath the name.

Martin Lerner  
Examiner  
Group Art Unit 2626